Blind Bin Picking of Small Screws Through In-finger Manipulation With Compliant Robotic Fingers

Matthew Ishige[1], Takuya Umedachi[1], Yoshihisa Ijiri[2], Tadahiro Taniguchi[3], and Yoshihiro Kawahara[1]

***Abstract*— Although picking up objects a few centimeters in size is a common task, achieving such ability in a robot manipulator remains challenging. We take a step toward solving this problem by focusing on the task of picking a 1.0-cm screw from a bulk bin using only tactile information to achieve the task. Inspired by how humans pick up small objects from a bin, we propose a "grasp-separate" strategy for robotic picking, which involves grasping many objects first and then separating a single object through manipulation in the fingers, for robotic picking. Based on this strategy, we developed a tactile-based screw bin-picking system. We trained a convolution neural network to estimate the number of screws in the fingers first and built a controller that generates manipulation behaviors to separate a screw using reinforcement learning. To compensate for the low resolution of off-the-shelf tactile sensor arrays, we adopted active sensing, which uses observations obtained during a predefined simple movement. We show that this approach enhances the estimation accuracy and manipulation performance. Furthermore, to enable flexible finger motion, such as between the thumb and the index finger in a human hand, we propose a soft robot finger structure that leverages compliant materials. A soft actor-critic algorithm successfully found dexterous screw separation behaviors in compliant soft robotic fingers. In the evaluation, the system obtained an average success rate of 80%, which was difficult to achieve without the grasp-separate manipulation technique.**

I. Introduction

Object picking is a fundamental task in robotic manipulation research and has long been investigated [1]. However, picking up objects a few centimeters in size is still challenging. Nevertheless, such ability is required for robots to perform many tasks in factory and home environments. Therefore, this paper takes a step toward the problem and focuses on picking a single screw from a bulk bin, which is a common task in product assembly.

Conventional robotic picking systems rely heavily on computer vision. However, it is difficult to adopt this approach to solve bin-picking tasks. One reason is that the occlusion problem is severe when a picking target is smaller than the fingertips. Another reason is that clear depth images, which are vital for vision-based object picking, are challenging to obtain for objects a few centimeters in size. Moreover, even if a camera can accurately obtain the initial configuration (i.e., position and angle) of a target object, it would be easily disturbed by the contact between the manipulator and the target. Therefore, picking systems based on other sensing modalities should be considered.

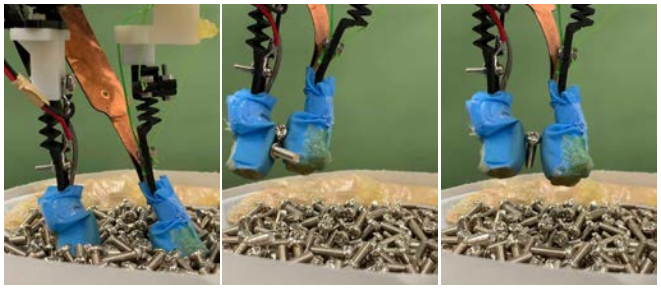


Fig. 1: "Grasp-separate" approach for blind screw bin picking. The blue fingertip portions are equipped with tactile sensors to recognize the number of target objects. The fingers initially pick up multiple screws and then manipulate them until one screw is left.

Tactile-based picking is a promising approach for bin-picking tasks. In this approach, a controller mainly relies on tactile and proprioceptive information. Previous works have achieved promising picking performance [2], [3]. Their strategy is to learn re-grasping policies, i.e., a controller learns how to update the grasping behavior on the basis of the current tactile observation. Although such a strategy is useful when target objects are placed apart, it is impractical to adopt the same strategy to the task of picking objects from a bulk bin because a manipulator can easily grasp multiple objects at once, and, thus, observations obtained in consecutive grasps are less relevant.

In this paper, we propose a tactile-based bin-picking approach through a "grasp-separate" strategy, which is inspired by how a human hand picks up a single screw from a bulk bin. When people conduct the task, they first grasp many screws at once and then they move their fingers to drop excess screws by relying on tactile feedback. We reproduced this behavior using a robot manipulator. To estimate the number of screws in the fingers and to determine the finger motions, we embedded compact pressure sensor arrays into the fingertips. For better estimation, we adopted active sensing, which obtains tactile observations under different finger positions. We also propose compliant soft robotic fingers that can easily perform fingertip surface sliding motion, akin to the motion between the thumb and the index finger in a human hand. The compliant fingers also simplify the grasping of screws. To harness compliance, which is known to be

*This work was supported by JST ERATO, Japan (grant number JPMJER1501).

[1] Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, Japan mishige, umedachi, kawahara@akg.t.u-tokyo.ac.jp

[2] OMRON SINIC X Corporation, 5-24-5 Hongo, Bunkyo-ku, Tokyo, Japan yoshihisa.ijiri@sinicx.com

[3] Department of Information Science and Engineering, Ritsumeikan University, 1-1-1 Noji-higashi, Kusatsu, Japan taniguchi@em.ci.ritsumei.ac.jp

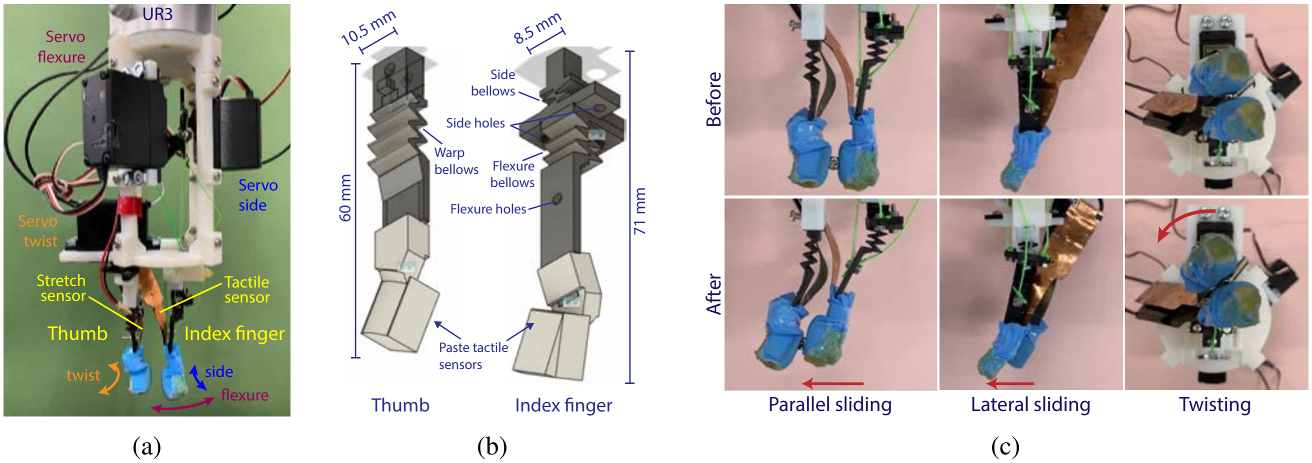


Fig. 2: Hardware design. (a) Overview of the manipulator. Fingers are driven by tendons controlled by three servomotors. The manipulator is attached to a UR3 robot arm. The fingertips are covered with blue silicon films. (b) Design of the thumb and index finger. Tendons are fixed to the holes using screws. The black parts are made of nylon, whereas the white parts (fingertips) are made of another plastic. Bellows-like structures function as flexure joints. (c) Three types of behaviors that compose the in-finger manipulation. The flexure of the index finger leads to parallel sliding motion, side motion of the index leads to lateral sliding, and rotation of the servomotor at the thumb base leads to twisting of the fingertip surface. These behaviors are combined during manipulation.

difficult to control in conventional rule-based manners [4], [5], we adopted a learning-based approach to fully leverage the flexibility in the proposed fingers.

The contributions of this study are as follows. First, we introduced the “grasp-separate” strategy for tactile-based bin picking and formulate it in a computational framework to achieve a stable performance. Second, we showed that active sensing is useful for overcoming the insufficient resolution of tactile sensors for both counting and manipulation. Third, we proposed a compliant manipulator that realizes human-like fingertip surface sliding motion with simple control and showed that it plays a vital role in separation manipulation. Finally, we showed that reinforcement learning is useful in finding proper behaviors in compliant hardware. We refer the readers to a supplemental video for the demonstration of the developed system.

II. Related Works

A. Tactile-Based Sensing and Active Sensing

Tactile sensing has long been investigated [6]. Among various tactile sensors, vision-based sensors [7], [8] have enabled robotic manipulators to achieve various tasks [9]–[13]. However, fingers with these sensors tend to be bulky owing to the embedded cameras and are not suitable for tasks in restricted spaces such as screw picking from a bin. On the other hand, capacitor- and resistor-based tactile sensor arrays measure pressure distribution and enable compact finger design. In a previous work, we developed a screw number estimation system using tactile data from capacitor-based pressure sensor arrays [14]. However, the fingers still suffer from partial observability when the screws in the fingers are piled (e.g., a tactile sensor in each finger can only touch one screw when two screws are crossing). To mitigate this problem, we leveraged a temporal observation sequence while moving the sensing parts. Such a strategy is called active sensing, and it is an actively investigated field [15]–[19]. We investigated whether this strategy leads to an accurate inference of the screw situation and to higher manipulation performance.

B. Tactile-Based Object Picking

Several studies utilized tactile information to predict grasp stability [20], [21]. According to the prediction, a grasping configuration was refined to achieve stable grasping. However, these methods require the position of a picking target object, which is normally obtained from vision. Murali et al. achieved high picking performance without vision and prior knowledge of the object position [22]. Their strategy is to train a re-grasping policy using reinforcement learning. Wu et al. adopted a similar strategy and improved the performance [3]. Unfortunately, this re-grasping strategy is not practical to apply when a manipulator has to pick up a small object from a bulk bin because (1) objects easily move away and (2) target objects are densely located such that a manipulator tends to grasp multiple objects at once. In particular, the second reason is difficult to mitigate even by adopting thin fingers. Therefore, we adopted a different strategy.

C. In-hand Manipulation and Compliant Manipulators

Although there are numerous literature studies on in-hand manipulation, realizing dexterous manipulations by human fingers (e.g., holding and adjusting the pose and location of tiny parts) still has a long way to go. Such ability is vital for robots to conduct tasks required in various environments, from houses to factories. We call them "in-finger manipulation." Previous works proposed underactuated manipulators that mimic human finger motion [23]–[26]. Although the mechanisms are sophisticated, they tend to be bulky and difficult to fit into a finger size. In addition, adding additional degrees of freedom (DoF) requires further complicated mechanisms (e.g., enabling sliding motion orthogonal to a manipulator is difficult). Restriction in finger motion is disadvantageous for dexterous in-finger manipulation. Zhou et al. proposed a robotic hand made of soft material for dexterous manipulation [27]. The material compliance simplified the tasks of moving an object in the fingers without dropping. Inspired by this idea, we propose a simple tendon-driven finger design that allows finger motion in multiple directions to achieve the screw-picking task.

III. Proposed System

A. Compliant Fingers

As shown in Fig. 2(a), the designed manipulator has two fingers: a thumb and an index finger. The fingers are driven by tendons attached to servomotors, as shown. The tactile sensor arrays described later are attached to the fingertips. The details of the finger design are presented in Fig. 2(b). The fingers are made of nylon and fabricated using a 3D printer (Markforged, Inc.). There are no joint assemblies in the fingers. Instead, bellows-like compliant structures function as "flexure joints." Because of this design, fingers can be batch molded, thus simplifying the fabrication process. The white parts in Fig. 2(b) are the removable fingertips made of plastic. Tactile sensor arrays, explained in the following section, are attached to these fingertips.

The index finger has two DoFs, namely, flexure motion and side motion. A tendon fixed to the "flexure hole" in Fig. 2(b) generates flexure motion (i.e., bending around the "flexure bellows"). Two tendons fixed to the "side holes" generate side motion (i.e., bending around the "side bellows"). The thumb has two DoFs, namely, twisting generated by a servomotor right under the thumb in Fig. 2(a) and warping around the "warp bellows." The latter DoF is entirely passive, i.e., warping occurs only when the index finger firmly presses the thumb. This passive compliance provides continuous pressure between the fingertips, which ensures that screws do not easily drop during manipulation. Furthermore, it is the key structure to enable parallel surface sliding motion, which is a vital manipulation behavior for screw separation. Note that the warp bellows do not allow a motion in the lateral direction. We empirically adjusted the thickness of the warp bellows (i.e., through trial and error). It should be hard enough to ensure sufficient pressure on the fingertip surface when the thumb is warping. On the other hand, it should be soft enough to realize the sliding motion with the adopted servo motor. The stiffness affects the picking performance, and this relationship should be investigated in future work. A flexible stretch sensor (Images Scientific Instruments, Inc.) was attached to the thumb to obtain the magnitude of the warping. The warp magnitude, together with the angle of the flexure servomotor, implicitly tells a controller the gap between the fingers, which conveys information about the screw configuration in the fingers. Figure 2(c) shows the following three manipulation behaviors generated by the compliant robotic fingers presented.

- **Parallel sliding**: The sliding motion of the fingertip surface in the parallel direction as the index finger presses the thumb to warp.
- **Lateral sliding**: The sliding motion of the fingertip surface in the lateral direction as the index finger generates side motion.
- **Twisting**: Generated by the twisting motion of the thumb. This motion leads to changes in the pressure distribution on the surface.

These motions are combined during in-finger manipulation (e.g., diagonal sliding can be realized by combining parallel and lateral sliding). Because the fingertip surface is rectangular owing to the shape of the tactile sensor, the range of lateral sliding was relatively small compared with that of parallel sliding. It should be investigated whether enlarging the range of lateral sliding improves the picking performance in future work.

We designed fingers focusing on an in-finger manipulator for bin-picking tasks. Hence, a simplified design was proposed. However, a similar compliance mechanism can be adopted in manipulators with higher DoFs to realize the same motions.

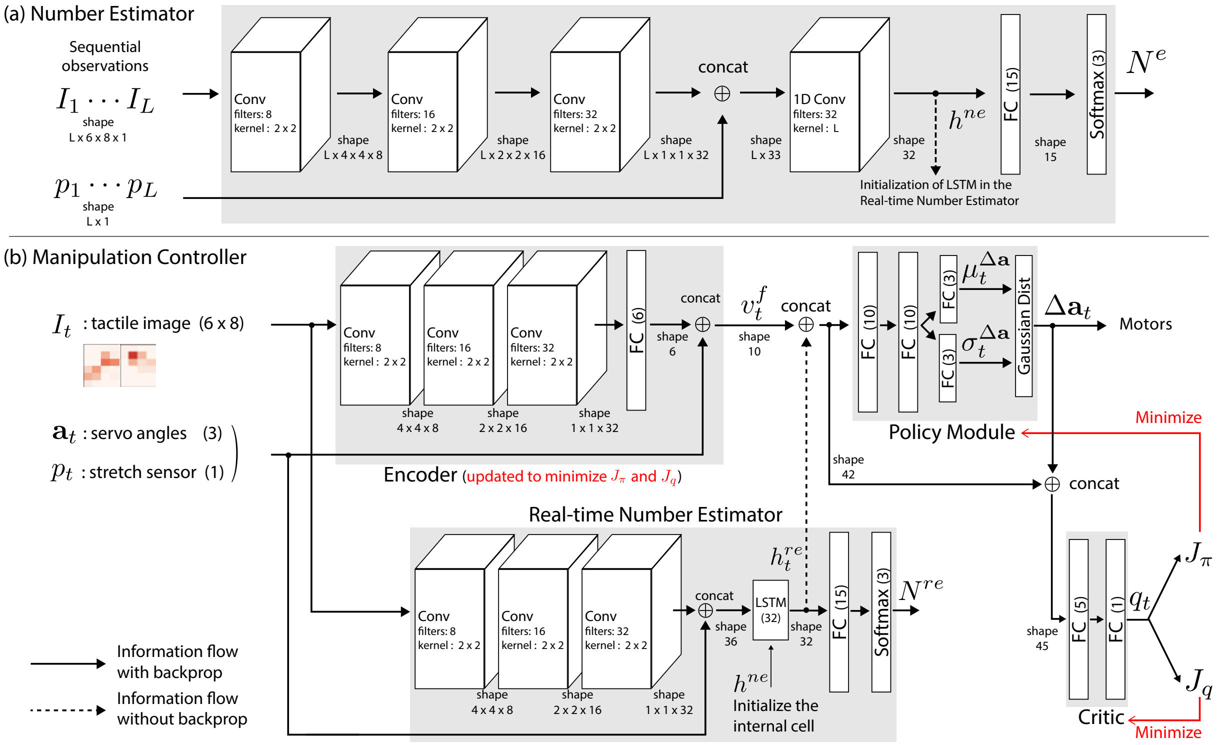
B. Tactile Sensor Array

We adopted a tactile sensor array model 4350 from the Pressure Profile System [28]. Its active sensing area is 12 mm × 8 mm, and there are 6×4 sensing elements. The size of each sensing element is 2 mm × 2 mm. The maximum sensing rate is 50 Hz. Because the raw values from the sensor are noisy, we took the average of the latest five consecutive raw images and regarded it as the latest sensor observation. A sensor array is attached to each fingertip; hence, two 6×4 size tactile images were obtained at every time step. Sample pressure images are shown at the left side of Fig. 3(b). The tactile images obtained were sent to a PC via Bluetooth. A sensor array is pasted on a fingertip pouch, and a thin sponge layer is attached to it. A whole fingertip (including a pouch, a sensor array, and a sponge) is covered with a thin silicon film to increase friction and protect the assembly.

C. Learning-based Control System

In this study, we implemented the "grasp-separate" approach as a consolidation of three successive processes, namely, random grasping, screw counting, and in-finger manipulation of screws. After a random grasp, the screw counting process and the in-finger manipulation process repeat until the system determines that a single screw is left in the grasp. If the system drops all screws accidentally, it starts over from the random grasp.

1) Random grasping: In this process, the manipulator dips the fingers into a screw bulk bin and pinches screws without any sensory feedback. The fingers tend to pick up multiple screws because of the compliant structure in the thumb and



(a) Number Estimator
Sequential observations
$I_1 \cdots I_L$
shape L x 6 x 8 x 1
$p_1 \cdots p_L$
shape L x 1
Conv filters: 8 kernel: 2 x 2
shape L x 4 x 4 x 8
Conv filters: 16 kernel: 2 x 2
shape L x 2 x 2 x 16
Conv filters: 32 kernel: 2 x 2
shape L x 1 x 1 x 32
concat
shape L x 33
1D Conv filters: 32 kernel: L
shape 32
h^{ne}
Initialization of LSTM in the Real-time Number Estimator
FC (15)
shape 15
Softmax (3)
N^e
(b) Manipulation Controller
I_t : tactile image (6 x 8)
$\mathbf{a}_t$: servo angles (3)
p_t : stretch sensor (1)
Conv filters: 8 kernel: 2 x 2
shape 4 x 4 x 8
Conv filters: 16 kernel: 2 x 2
shape 2 x 2 x 16
Conv filters: 32 kernel: 2 x 2
shape 1 x 1 x 32
FC (6)
shape 6
concat
v_t^f
shape 10
concat
Encoder (updated to minimize J_π and J_q)
FC (10)
FC (10)
FC (3)
FC (3)
$\mu_t^{\Delta \mathbf{a}}$
$\sigma_t^{\Delta \mathbf{a}}$
Gaussian Dist
$\Delta \mathbf{a}_t$
Motors
Policy Module
Minimize
shape 42
concat
Real-time Number Estimator
Conv filters: 8 kernel: 2 x 2
shape 4 x 4 x 8
Conv filters: 16 kernel: 2 x 2
shape 2 x 2 x 16
Conv filters: 32 kernel: 2 x 2
shape 1 x 1 x 32
concat
shape 36
LSTM (32)
h_t^{re}
shape 32
FC (15)
Softmax (3)
N^{re}
h^{ne}
Initialize the internal cell
shape 45
FC (5)
FC (1)
q_t
J_π
J_q
Critic
Minimize
Information flow with backprop
Information flow without backprop

Fig. 3: The overview of computational models. (a) The computational model for estimating the number of screws. Inputs to the number estimator are tactile images $I_1 \cdots I_L$ and L stretch sensor values $p_1 \cdots p_L$, which are generated from predefined L steps movement of the fingers. Each image in $I_1 \cdots I_L$ is processed by convolution layers, respectively. The outcome, which has a size of $L \times 1 \times 1 \times 32$channels, is concatenated with $p_1 \cdots p_L$ and is processed by a 1D convolution layer followed by a fully connected layer. Finally, softmax operation is applied to obtain the probability of screws being zero, one, or multiple. Numbers under "shape" signature indicate the size of an output tensor from a previous layer. "FC" represents a fully connected layer, and a number following it is the number of its units. (b) The computational model for the in-finger manipulation controller. A manipulation process lasted for a fixed step (it was set to five in the following evaluations). A set of observations at each step (I_t, a_t, p_t) is given to the encoder to obtain a feature vector v_t^f. It is also fed into the real-time number estimator, which outputs an estimation of the current screw number using an LSTM layer. The LSTM layer has 32 hidden units. The feature vector v_t^f is concatenated with a hidden layer of the real-time number estimator h_t^{re} and fed into the policy module to obtain the next motion angles $\Delta \mathbf{a}_t$ of the servomotors. For activation function, Tanh is used in the LSTM and ReLU is used in all the other layers presented here.

the sponges at the fingertips. To drop screws incidentally caught by the finger edges, which is almost impossible to be recognized by the pressure sensors, the UR3 robot arm shakes the manipulator once.

2) Screw counting process: This process conducts active sensing to obtain sequential observations and estimates the number of screws on the basis of the observations. The computational model for the estimation is depicted in Fig. 3(a). The system moves the fingers in L steps in a predefined manner (e.g., flexure of the index finger in L steps) while recording the observations. An observation O_t is a set of tactile images I_t (two 6×4 arrays from the fingers are concatenated to form a 6×8 array), the current servomotor angles $\mathbf{a}_t = (a_t^{flex}, a_t^{side}, a_t^{twist})$, and the stretch sensor value p_t. A sequence of angles $\mathbf{a}_{l,\cdots,L}$ is fixed, and, thus, it is excluded from the input to the number estimator. It outputs the probabilities of the number being either zero, one, or multiple. The parameters in the estimator are updated to minimize an estimation error from the actual number using cross-entropy loss. The estimator is pretrained before training a manipulation controller, and its parameters are fixed afterwards.

3) In-finger manipulation process: This process separates a single screw from the grasp of multiple screws through in-finger manipulation at fixed steps. The computational model for the manipulation controller is depicted in Fig. 3(b). The policy module controls the fingers by outputting the next move angles of the servomotor $\Delta \mathbf{a}_t = (\Delta a_t^{flex}, \Delta a_t^{side}, \Delta a_t^{twist})$ on the basis of the observations at each step. An observation O_t is fed into both the encoder and the real-time number estimator. The encoder converts O_t into a feature vector v_t^f through convolution layers. The real-time number estimator is a long short-term memory (LSTM)-based screw counter that estimates the number using the observations from the beginning of a manipulation process. The output of the encoder v_t^f and a hidden layer value of the real-time number estimator h_t^{re} are concatenated and fed into the policy module. The policy network calculates means and standard deviations of a Gaussian distribution from which the next motion $\Delta\mathbf{a}_t$ is sampled. The hidden layer of the real-time number estimator h_t^{re} is provided to the policy because it is likely to contain richer information on the screws in the fingers. The encoder and the policy module are trained by the soft actor-critic (SAC) algorithm [29]. SAC searches for the following optimal policy:

$$\pi^* = \arg\max_{\pi} \sum_t \mathbb{E}_{(O_t,\Delta\mathbf{a}_t)\sim\rho_\pi}\left[r(O_t,\Delta\mathbf{a}_t) + \alpha\mathscr{H}(\pi(\cdot|O_t))\right] \tag{1}$$

where ρ_π is a distribution over pairs of an observation and an action, r is a reward function, $\mathscr{H}(\pi(\cdot|O_t))$ is the entropy of the action distribution given an observation O_t, and α is a temperature parameter that determines the relative importance of the entropy term. The reward function is defined as,

$$r(O_t,\Delta\mathbf{a}_t) = \begin{cases} 0 & \text{(multiple screws left)} \\ 1 & \text{(one screw left)} \\ -1 & \text{(no screws left)} \end{cases} \tag{2}$$

To obtain the optimal policy, we need to prepare a critic network, which approximates the Q-value, and to alternately update the policy and the critic using the collected samples during the training process. The critic is updated to minimize the following objective:

$$\begin{aligned} J_Q &= \mathbb{E}_{(O_t,\Delta\mathbf{a}_t)\sim\mathscr{D}}\left[\frac{1}{2}\left(Q(O_t,\Delta\mathbf{a}_t) - \bar{Q}\right)^2\right] \qquad (3) \\ \bar{Q} &= r_t + \gamma\, \mathbb{E}_{O_{t+1}\sim p,\Delta\mathbf{a}_{t+1}\sim\pi}\left[Q(O_{t+1},\Delta\mathbf{a}_{t+1})\right] \end{aligned}$$

where $r_t = r(O_t,\Delta\mathbf{a}_t)$, $(O_t,\Delta\mathbf{a}_t) \sim \mathscr{D}$ represents drawing samples from an experience replay buffer and γ is a discount parameter. The policy is updated to minimize the following objective:

$$J_\pi = \mathbb{E}_{O_t\sim\mathscr{D},\Delta\mathbf{a}_t\sim\pi}\left[\alpha\log(\pi(\mathbf{a}_t|O_t)) - Q(O_t,\Delta\mathbf{a}_t)\right] \tag{4}$$

The encoder is updated by both J_Q and J_π because v_t^f is fed into the policy and critic. We adopted other implementation details presented in the original paper [29], such as the automatic adjustment of α and double Q-networks. The real-time number estimator is trained independently, and the parameters are updated to minimize the estimation error using cross-entropy loss. The internal cell of the LSTM in the estimator is initialized using h^{ne} from the number estimator during the counting process.

IV. Experiment

A. Hardware Setups

The hand mentioned above was attached to the UR3 robot arm. An Arduino UNO controlled both the servomotors and the UR3 robot arm. An electrical scale was prepared, and a bin of screw bulk was placed on it to monitor the number of screws being picked up. This screw number was used in the training of the number estimator and the real-time number estimator. It was also used to calculate a reward in reinforcement learning. The tactile sensor arrays, Arduino UNO, and electrical scale were connected to a Windows 10-based PC and controlled by a Python script. The following experiments were conducted using these setups. The diameter and length of the target screws were 3.0 mm and 10.0 mm, respectively.

TABLE I: COMPARISON OF ESTIMATION ACCURACY WITH DIFFERENT ACTIVE SENSING TYPES

One observation	Flexure	One observation*
0.750 (±0.064)	0.833 (±0.047)	0.800 (±0.062)

B. Screw Number Estimation Accuracy

This section presents an evaluation result of the active sensing accuracy. We compared the accuracy of the screw number estimation under different sensing conditions.

- **One observation**: Numbers were estimated using a single observation after random picking.
- **Flexure**: Numbers were estimated using sequential observations obtained during the flexure motion of the index finger. The servomotor for flexure motion was turned for 8° to press the thumb first, turned for 10° to the opposite side to loosen, and, finally, turned for 2° to return to the home position. Hence, the length of an observation sequence is $L = 3$ in this setting.
- **One observation***: Numbers were estimated using the last observation obtained after the flexure active sensing motion mentioned above. The length of an observation sequence is $L = 1$ in this setting.

We collected 510 training data (zero: 70, one: 170, and multiple: 270) and 90 test data (zero: 30, one: 30, and multiple: 30) for one trial.

Table I shows the estimation accuracy of the conditions mentioned above. Each result is an average of three models trained independently with different data sets. A comparison between the results of the **flexure** active sensing and **one observation*** indicates that using sequential observations is useful to raise the accuracy. A comparison between **one observation** and **one observation*** shows that the flexure motion itself contributed to the estimation accuracy. The following experiments were conducted using flexure active sensing.

C. Picking Performance

This section presents the evaluation results of the picking performance. We conducted two evaluations, namely, **separation performance** evaluation, which focused on a manipulation policy, and **picking performance** evaluation, which considered the total system.

In this experiment, the ranges of servomotor angles $\mathbf{a} = (a^{flex}, a^{side}, a^{twist})$ were

$$a^{flex} \in [-2°, 20°], \quad a^{side} \in [-2°, 2°], \quad a^{twist} \in [-12°, 12°] \tag{5}$$

where $\mathbf{a} = \mathbf{0}$ is the home position (a configuration right after random grasping). A manipulation process lasted for four

TABLE II: SCREW PICKING PERFORMANCE EVALUATION RESULTS

	Separation performance		Picking performance	
	Success rate	Per cycle	Success rate	Per pickup
No manipulation	–	–	0.47 (±0.339)	0.31 (±0.23)
Random manipulation	0.77 (±0.05)	0.44 (±0.05)	0.77 (±0.05)	0.53 (±0.08)
Proposed method	**0.93** (±0.05)	**0.70** (±0.02)	**0.80** (±0.08)	**0.78** (±0.06)
Proposed method without active sensing	0.80 (±0.08)	0.50 (±0.13)	–	–
Proposed method without the real-time number estimator	0.90 (±0.08)	0.52 (±0.09)	–	–

steps. The policy module output $\Delta\mathbf{a}_t$ is restricted as

$$\Delta a_t^{flex} \in [-5,5], \quad \Delta a_t^{side} \in [-1,1], \quad \Delta a_t^{twist} \in [-3,3] \qquad (6)$$

Although it was a continuous vector originally, it was converted into integers and was multiplied by $(2,2,4)$ element-wise to restrict the action space. Hence, the actual motor command $\Delta\hat{\mathbf{a}}_t = (\Delta\hat{a}_t^{flex}, \Delta\hat{a}_t^{side}, \Delta\hat{a}_t^{twist})$ is

$$\begin{aligned} \Delta\hat{a}_t^{flex} &\in \{\, 2\times i \mid i\in\mathbb{Z},\ -5\le i\le 5 \,\} \\ \Delta\hat{a}_t^{side} &\in \{\, 2\times i \mid i\in\mathbb{Z},\ -1\le i\le 1 \,\} \\ \Delta\hat{a}_t^{twist} &\in \{\, 4\times i \mid i\in\mathbb{Z},\ -3\le i\le 3 \,\} \end{aligned} \qquad (7)$$

We compared the proposed method with the following baselines:

- **No manipulation**: The manipulation process was not allowed, and it repeated the random grasping and number estimation until the number estimator had determined that the screw separation succeeded.
- **Random manipulation model**: Motion angles $\Delta\mathbf{a}_t$ were sampled uniformly at random from the range mentioned above during the in-finger manipulation.
- **Proposed method without active sensing**: Only a single observation was fed into the number estimator in this baseline.
- **Proposed method without the real-time number estimator**: The real-time number estimator was removed, and the hidden layer value h^{ne} was used instead of h_t^{re}.

The flexure active sensing motion (see SECTION IV-B) was adopted in this evaluation. The number estimators were pretrained for 500 epochs using data from 550 active sensing samples. After the pretraining, we trained each policy for 200 epochs.

1) Separation performance: This evaluation focused on the performance of dropping excessive screws to separate a single screw. A trial started with multiple screws in the fingers. The manipulator was allowed to repeat the manipulation cycles up to 10 times. If the manipulator succeeded in separating a screw within this limitation, a trial was counted as successful. It was labeled as failed if the manipulator dropped all the screws. We repeated this trial 10 times and calculated the success rate.

The “separation performance column of Table II shows the average separation performance of three independently trained controllers. The proposed method showed a higher separation success rate than that of random manipulation. We also presented the success per cycle:

$$\begin{cases} \frac{1}{\#\,\text{manipulation cycles until separation}} & \text{(if success)} \\ 0 & \text{(else)}, \end{cases} \qquad (8)$$

This value should be higher if a manipulation policy can quickly drop excess screws. The proposed method showed the highest success per cycle, compared with those of its variants, indicating the importance of the components.

2) Picking performance: This evaluation focused on the integrated performance of screw counting and separation in-finger manipulation and shows how the proposed system effectively picks up a single screw. In a trial, the manipulator first conducted random grasping and repeated screw counting and in-finger manipulation alternately up to 10 times. If the number estimator determined that the separation was not successful after 10 manipulation cycles or had dropped all the screws, the manipulator executed the random grasping again. A trial was terminated as soon as the number estimator determined that the separation was successful. Random picking was allowed up to five times. A trial was considered successful only if the number estimator had determined that the separation was successful, and, actually, it was. We evaluated the **no manipulation** model, the **random manipulation** model, and the **proposed** model.

The “picking performance” column of Table II shows the averaged manipulation performance of three trials. We presented the averages of the success rate and averages of the success per pickup:

$$\begin{cases} \frac{1}{\#\,\text{pick up}} & \text{(if success)} \\ 0 & \text{(else)}, \end{cases} \qquad (9)$$

Our system showed a higher performance than those of the baselines.

D. Qualitative Evaluation of Manipulation Behaviors

This section presents the typical manipulation behaviors shown by the manipulator. In Fig. 4(a), the manipulator successfully separated a screw from two screws mainly using parallel sliding. During the manipulation, the index finger firmly pressed the thumb and generated a large parallel sliding motion. As a result, a screw on the tip dropped and the other remained in the finger. This example emphasizes the importance of the compliant structure. This behavior is realized by the compliant structure of the proposed fingers.

Another successful manipulation instance is presented in Fig. 4(b). In this case, screws were aligned orthogonal to the direction of the fingers. Thus, the parallel sliding presented does not contribute when dropping only one screw. Instead, the manipulator used lateral sliding. These cases show that the policy adequately used different separation strategies according to the screw situation.

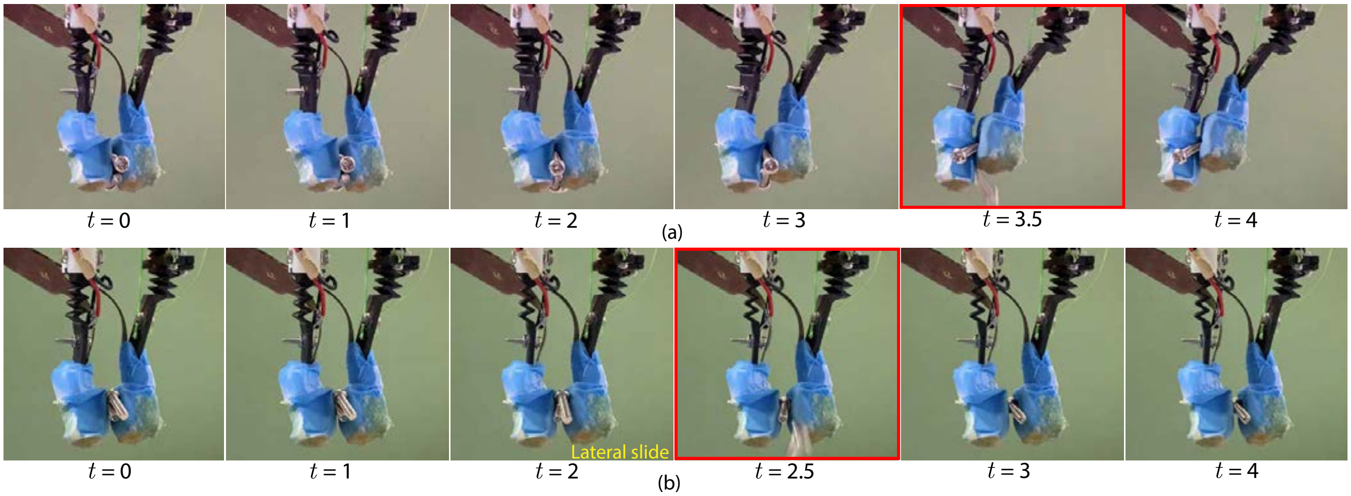


Fig. 4: Snapshots of the successful manipulation behaviors. Sequences of finger images at each step are shown. A step number is indicated by t for each image. The moments when excess screws drop are indicated by red boxes. (a) Separation where parallel sliding contributed. The fingers dropped excessive screws at $t = 3.5$ (between the third and the fourth step). (b) Separation where lateral sliding contributed. The fingers dropped excessive screws at $t = 2.5$ (between the second and the third step).

V. Discussion

A. Utility of Active Sensing

The evaluation result in SECTION IV-B confirmed that active sensing contributes to the number estimation accuracy. We suspect two reasons for this result. First, multiple observations obtained from different finger configurations compensated for the partial observability of the tactile images. For example, when screws in the fingers are piled, the fingertip surface cannot fully make contact with all the screws, which usually leads to incorrect estimation. However, there is a higher chance of touching such screws during an active motion. It is especially useful for distinguishing between single and multiple screws. This is probably why **flexure** active sensing yielded better estimation accuracy than those of both **one observation** and **one observation***. Second, the motion for active sensing reduced the diversity of screw alignments. We observed cases where active sensing motion itself dropped screws that were caught at the top and bottom edges of the fingertips, which is difficult to count because of the low sensitivity in sensor edges. In some cases, a screw perpendicular to the fingertip surface transformed parallel to the surface after the motion. Such an effect was likely to ease the estimation. This explains the difference in estimation accuracy between **one observation** and **one observation***. It is a type of sensorimotor coordination [30].

B. Importance of Active Sensing and Real-Time Number Estimator

Although the proposed method and its variants (one without active sensing and one without real-time number estimator) showed a similar success rate in separation performance, the variants were inferior to the full model in separation efficiency (i.e., success rate per manipulation cycle). This result indicates that both components are vital for realizing effective in-finger manipulation. The overall performance without active sensing, however, was lower than that without the real-time number estimator. This is probably because the initial inference of the screw configuration was less accurate in the model without active sensing. Because a manipulation cycle consists only of four steps, failure in the initial inference had a more significant impact on the performance than the lack of the real-time screw-counting component. Prolonging the manipulation duration may change the relative importance of active sensing and the real-time number estimator.

VI. Conclusions

This paper proposed a tactile information-based method that can perform the task of picking a $1.0-$ screw from a bulk bin through a grasp-separate strategy, which is inspired by the way a human hand conducts the task. The method consists of three processes: 1) a random grasping process wherein a manipulator pinches multiple screws, 2) a screw-counting process wherein the system estimates the screw number by processing tactile images from fingertips with a convolution neural network, and 3) in-finger manipulation process wherein a manipulator drops excessive screws to separate one screw. The screw-number estimator is trained to minimize the estimation error, and the manipulation controller is trained to leave only one screw in the fingers. The method achieved efficient picking performance compared with merely re-grasping until success was reached. In addition, we showed that adopting active sensing to mitigate the partial observability of the tactile sensor arrays contributed to both the estimation accuracy and the manipulation performance. Furthermore, we showed that flexible behaviors realized by the compliant finger design and automatic behavior exploration using reinforcement learning played a critical role in the screw separation process.

There are several directions for further development of the method. The applicability of this method to other types of objects should be tested. A thorough study of how each component of the method contributes to the final performance (e.g., is the stretch sensor in the thumb important?) is needed as well. It is also vital to seek durable materials for the fingers to make the system durable so that it can be used in production lines. Exploring reinforcement learning methods other than SAC is also interesting. In particular, methods that use latent representations constructed with temporal information using probabilistic inference [31], [32] are likely to improve the performance because screw alignments are probabilistic. Exploring sensing modalities other than the normal force used in this work is another exciting research direction. The human finger senses considerably richer information to infer much about the objects in the fingers. Integrating multiple modalities may lead to picking without retraining, i.e., picking unseen objects without additional parameter tuning. Such ability will significantly broaden the application of robot manipulators.

Acknowledgment

This work was supported by JST ERATO (grant no. JPMJER1501).

References

[1] G. Du, K. Wang, and S. Lian, "Vision-based robotic grasping from object localization, pose estimation, grasp detection to motion planning: A review," *arXiv preprint arXiv:1905.06658*, 2019.

[2] E. Johns, S. Leutenegger, and A. J. Davison, "Deep learning a grasp function for grasping under gripper pose uncertainty," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2016, pp. 4461–4468.

[3] B. Wu, I. Akinola, J. Varley, and P. Allen, "Mat: Multi-fingered adaptive tactile grasping via deep reinforcement learning," *arXiv preprint arXiv:1909.04787*, 2019.

[4] T. G. Thuruthel, E. Falotico, F. Renda, and C. Laschi, "Model-based reinforcement learning for closed-loop dynamic control of soft robotic manipulators," *IEEE Transactions on Robotics*, vol. 35, no. 1, pp. 124–134, 2018.

[5] M. Ishige, T. Umedachi, T. Taniguchi, and Y. Kawahara, "Exploring behaviors of caterpillar-like soft robots with a central pattern generator-based controller and reinforcement learning," *Soft Robotics*, vol. 6, no. 5, pp. 579–594, 2019.

[6] L. Zou, C. Ge, Z. J. Wang, E. Cretu, and X. Li, "Novel tactile sensor technology and smart tactile sensing systems: A review," *Sensors*, vol. 17, no. 11, p. 2653, 2017.

[7] W. Yuan, S. Dong, and E. Adelson, "Gelsight: High-resolution robot tactile sensors for estimating geometry and force," *Sensors*, vol. 17, no. 12, p. 2762, 2017.

[8] A. Yamaguchi and C. G. Atkeson, "Implementing tactile behaviors using fingervision," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics*. IEEE, 2017, pp. 241–248.

[9] S. Dong, W. Yuan, and E. H. Adelson, "Improved gelsight tactile sensor for measuring geometry and slip," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2017, pp. 137–144.

[10] J. Li, S. Dong, and E. H. Adelson, "End-to-end pixelwise surface normal estimation with convolutional neural networks and shape reconstruction using gelsight sensor," in *2018 IEEE International Conference on Robotics and Biomimetics*. IEEE, 2018, pp. 1292–1297.

[11] Y. She, S. Wang, S. Dong, N. Sunil, A. Rodriguez, and E. Adelson, "Cable manipulation with a tactile-reactive gripper," *arXiv preprint arXiv:1910.02860*, 2019.

[12] J. Lin, R. Calandra, and S. Levine, "Learning to identify object instances by touch: Tactile recognition via multimodal matching," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3644–3650.

[13] S. Tian, F. Ebert, D. Jayaraman, M. Mudigonda, C. Finn, R. Calandra, and S. Levine, "Manipulation by feel: Touch-based control with deep predictive models," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 818–824.

[14] M. Ishige, T. Umedachi, Y. Ijiri, and Y. Kawahara, "In-hand small-object counting from tactile sensor arrays installed on soft fingertips," in *2020 3rd IEEE International Conference on Soft Robotics*. IEEE, 2020, pp. 272–277.

[15] S. S. Baishya and B. Bäuml, "Robust material classification with a tactile skin using deep learning," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2016, pp. 8–15.

[16] J. A. Fishel and G. E. Loeb, "Bayesian exploration for intelligent identification of textures," *Frontiers in Neurorobotics*, vol. 6, p. 4, 2012.

[17] S. Funabashi, S. Morikuni, A. Geier, A. Schmitz, S. Ogasa, T. P. Torno, S. Somlor, and S. Sugano, "Object recognition through active sensing using a multi-fingered robot hand with 3d tactile sensors," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2018, pp. 2589–2595.

[18] N. F. Lepora and B. Ward-Cherrier, "Superresolution with an optical tactile sensor," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2015, pp. 2686–2691.

[19] L. Cramphorn, B. Ward-Cherrier, and N. F. Lepora, "Tactile manipulation with biomimetic active touch," in *2016 IEEE International Conference on Robotics and Automation*. IEEE, 2016, pp. 123–129.

[20] H. Dang and P. K. Allen, "Stable grasping under pose uncertainty using tactile feedback," *Autonomous Robots*, vol. 36, no. 4, pp. 309–330, 2014.

[21] Y. Chebotar, K. Hausman, Z. Su, G. S. Sukhatme, and S. Schaal, "Self-supervised regrasping using spatio-temporal tactile features and reinforcement learning," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2016, pp. 1960–1966.

[22] A. Murali, Y. Li, D. Gandhi, and A. Gupta, "Learning to grasp without seeing," *arXiv preprint arXiv:1805.04201*, 2018.

[23] W. G. Bircher, A. M. Dollar, and N. Rojas, "A two-fingered robot gripper with large object reorientation range," in *2017 IEEE International Conference on Robotics and Automation*. IEEE, 2017, pp. 3453–3460.

[24] N. Rojas, R. R. Ma, and A. M. Dollar, "The gr2 gripper: an underactuated hand for open-loop in-hand planar manipulation," *IEEE Transactions on Robotics*, vol. 32, no. 3, pp. 763–770, 2016.

[25] M. Liarokapis and A. M. Dollar, "Deriving dexterous, in-hand manipulation primitives for adaptive robot hands," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2017, pp. 1951–1958.

[26] B. Ward-Cherrier, N. Rojas, and N. F. Lepora, "Model-free precise in-hand manipulation with a 3d-printed tactile gripper," *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 2056–2063, 2017.

[27] J. Zhou, J. Yi, X. Chen, Z. Liu, and Z. Wang, "Bcl-13: A 13-dof soft robotic hand for dexterous grasping and in-hand manipulation," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3379–3386, 2018.

[28] Pressure profile systems. [Online]. Available: https://pressureprofile.com/

[29] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel, *et al.*, "Soft actor-critic algorithms and applications," *arXiv preprint arXiv:1812.05905*, 2018.

[30] R. Pfeifer and J. Bongard, *How the body shapes the way we think: a new view of intelligence*. MIT press, 2006.

[31] D. Hafner, T. Lillicrap, I. Fischer, R. Villegas, D. Ha, H. Lee, and J. Davidson, "Learning latent dynamics for planning from pixels," *arXiv preprint arXiv:1811.04551*, 2018.

[32] A. X. Lee, A. Nagabandi, P. Abbeel, and S. Levine, "Stochastic latent actor-critic: Deep reinforcement learning with a latent variable model," *arXiv preprint arXiv:1907.00953*, 2019.